Re 09/241,823 -- EAST 1.1

Type	L #	Hits	Search Text	DBs	Time Stamp
IS&R	L1	1	("5860012").PN.	USPAT	2001/01/19 13:32
BRS	L2	9	5860012.URPN.	USPAT	2001/01/19 13:23
BRS	L3	7	5021949.PN. OR 5086502.PN. OR 5257378.PN. OR 5265239.PN. OR 5361358.PN. OR 5386564.PN. OR 5388211.PN.	USPAT	2001/01/19 13:30
IS&R	L4	1	("5860012").PN.	USPAT	2001/01/19 13:39
IS&R	L5	1	("6006034").PN.	USPAT	2001/01/19 14:03
BRS	L6	2	6006034.URPN.	USPAT	2001/01/19 13:40
BRS	L7	10	4712214.PN. OR 5267235.PN. OR 5396613.PN. OR 5495610.PN. OR 5553083.PN. OR 5583563.PN. OR 5721824.PN. OR 5835724.PN. OR 5974461.PN. OR 6006034.PN.	USPAT	2001/01/19 13:42
BRS	L8	26	4714992.PN. OR 5005122.PN. OR 5008814.PN. OR 5019963.PN. OR 5155847.PN. OR 5247683.PN. OR 5343527.PN. OR 5440723.PN. OR 5448727.PN. OR 5495610.PN. OR 5555416.PN. OR 5577244.PN. OR 5581764.PN. OR 5586304.PN. OR 5721911.PN. OR 5724345.PN. OR 5732266.PN. OR 5732275.PN. OR 5734898.PN. OR 5737533.PN. OR 5752042.PN. OR 5754830.PN. OR 5758342.PN. OR 5768511.PN. OR 5794259.PN. OR 5848421.PN.	USPAT	2001/01/19 13:47
BRS	L9	23	5752042.URPN.	USPAT	2001/01/19 13:48
BRS	L10	186241	(install$5 or download$3 or down adj load$3)and automatic$4	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 14:19
BRS	L11	39479	updat$3 or upgrad$3)and(periodic$4 or timeout or time adj out or time$1 adj event	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 14:45
BRS	L12	9854	l10 and l11	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 14:46
BRS	L13	85	restor$3 same ((first or old)near5 version)	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 14:48
BRS	L14	10	l12 and l13	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 14:49
BRS	L15	13136	updat$3 or upgrad$3)same(periodic$4 or timeout or time adj out or time$1 adj event	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 15:53
BRS	L16	3167	l10 and l15	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 15:54
BRS	L17	75410	((less than)or(at least))near5(hour$1 or day$1)	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 15:58
BRS	L18	47	l17 same l15	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 15:58
BRS	L19	13	l18 and l10	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 16:17
IS&R	L20	472	("717/11").CCLS.	USPAT	2001/01/19 16:17
IS&R	L22	1212	("713/100,153,163,187,191,200").CCLS.	USPAT	2001/01/19 16:24
IS&R	L23	1824	("709/203,201,2221,220,227").CCLS.	USPAT	2001/01/19 16:28
IS&R	L24	1887	("707/203,202,204,10").CCLS.	USPAT	2001/01/19 16:28
IS&R	L25	247	("714/13,18").CCLS.	USPAT	2001/01/19 16:34
IS&R	L26	1069	("340/541,540").CCLS.	USPAT	2001/01/19 16:37
BRS	L27	219	l16 and l17	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 16:40
BRS	L28	1	l27 and l20	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 16:40
IS&R	L29	654	("705/51,54,26").CCLS.	USPAT	2001/01/19 16:45
BRS	L30	4	l27 and l29	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 16:48
BRS	L31	12	l27 and (l22 or l23 or l24 or l25 or l26)	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 17:19
BRS	L32	4	l16 and l26	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 16:58
BRS	L33	267	l16 and (l20 or l22 or l23 or l24 or l25)	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 17:20
BRS	L34	3	l13 and l33	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 17:22
BRS	L35	12	l17 and l33	USPAT; EPO; JPO; Derwent; IBM TDB	2001/01/19 17:26

DOCUMENT-IDENTIFIER: US 6154878 A
TITLE: System and method for on-line replacement of software

ABPL:
A system for changing a software library during the execution of a software application using the software library. The software application interfaces to the software library only through the use of an interface library, to ensure that the software application does not directly bind with the software library. With no direct binding the software library can be updated during runtime without the software application re-resolving the location of the software library. The update is triggered by a change of the version number in a registry. The program correctness is maintained by library management services ensuring that the software library is no longer in use by the application before updating to the new library. Memory management services are used to ensure that the state of the library is maintained between the old and the updated versions of the software library.

BSPR:
This invention relates generally to reducing down-time during software replacement and more particularly to the automatic dynamic updating of software.

BSPR:
In operation, whenever the application invokes a routine found in the implementation library, control transfers to the proxy of the routine in the interface library. Before calling the actual routine in the implementation library, the interface library first checks the registry to see if the implementation library is due to be updated . If so updated, the library management services ensure that the implementation library is able to handle the change at this moment. When the implementation library is ready, (i.e., not in use) it is swapped out for its new version. The new version will restore the state of the old version using the memory management services and is then ready for use. The interface library then calls the updated routine in the new implementation library.

BSPR:
This system allows the application to employ the new implementation library at a time when it is convenient to the application. Prior art systems force the application to update at a specific time once the new library is ready. The use of a registry to signal to the application that a new version is ready allows the update to occur, not when the library is first updated, but at the next time the new library is accessed.

DEPR:
Another variation on this calling convention is to divide a "do forever" loop, e.g. a call-back loop, between the interface library and the implementation library. If the implementation library contains functionality that waits for some external event, such as a mouse click, the implementation library can periodically check to see if it is still the current version of the library. The interface library may contain a simple loop to check the registry to see if the running version of the implementation library is the current version. As long as the running implementation library remains the current version, the loop in the interface library will call the running version of the implementation library to perform the appropriate tasks. When a change of the implementation library is requested, the interface library will detect the change the next time it calls the implementation library. Before calling the implementation library, the interface library will invoke the library management services to update to the new version of the implementation library.

DEPR:
Data storage in an implementation library must ensure that the state of the library is preserved between the old and the new version of the library. Requiring the implementation library to maintain a strict memory management discipline allows for the revision of the implementation library to be invisible to the application. The entire data state of the library will be restored when the library is updated reflecting all changes to the data made by the previous version of the library.

DOCUMENT-IDENTIFIER: US 6003013 A
TITLE: Customer worth differentiation by selective activation of physical instrumentalities within the casino

ABPL:
A system and method for differentiating customers according to their worth to the casino. Customer information is accumulated at each affiliated casino through one or more LAN-based management systems, updated to a central patron database (CPDB) that is coupled to each casino LAN through a WAN, and made available to each affiliated casino property as needed. Customer accounts are automatically activated and provided with data from the CPDB when a customer from one casino property first visits an affiliated casino property. Customer accounts are updated with status information based on the customer's worth to the casino. Customer accounts are updated with new activity data whenever a management system associated with the casino receives customer data from input devices, such as card readers, workstations, and dumb terminals, located at various venues throughout the casino.

DEPR:
CPDB updates 460 are required to synchronize data in the embodiment of system 100 employing distributed CMSs 234 and centralized CMS 284 (FIGS. 2A and 2B, and FIG. 2C, respectively). In the former configuration, these updates periodically transfer accumulated data from all accounts in local management systems (typically, CMS 234) to CPDB 220. In the preferred embodiment of this configuration, updates to CPDB 220 are scheduled at least once every twenty four hours at time periods when activities on casino LANs 120, central LAN 110, and WAN 102 are low. Where WAN 102 has a high bandwidth, data updates may be made without regard to other traffic on WAN 102. For the reasons discussed above, CPDB updates from the data store of central CMS 284 do not impact data traffic on WAN 102 and may be scheduled more flexibly.

DOCUMENT-IDENTIFIER: US 5999947 A
TITLE: Distributing database differences corresponding to database change events made to a database table located on a server computer

BSPR:
One way to distribute changes made to a database or database table is to download the entire table each time a client makes a connection with the server. While practical when a data store is relatively small, a larger database or database table will require large amounts of bandwidth on the communications link. This will make for an expensive and time consuming transfer that, in many instances, will be intolerable and impracticable.

BSPR:
What is generally sought in database change distribution systems described above are ways to quickly send the minimum amount of information needed to update a remote data store. This allows the client to quickly make a connection with the server, download only the necessary and sufficient amount of information, and make changes to the client copy of the data store without expending undue time or computing resources.

DEPR:
One example of an intermittent connection environment would be the servicing example explained previously. In that environment, a parts database is centrally managed and updated but is used by field service representatives
having laptop computers (i.e., clients). The field service representatives will only intermittently connect with the home office server computer on a periodic and often random basis ranging from a couple of times per day to weekly or even less frequently.

DEPR:

At step 78 (FIG. 5), it is determined whether a major revision or conversely a minor revision is chosen for the update creation. A major revision may be indicated manually, automatically after so many updates have been made, based on significant structural changes made to the table requiring a re-copy of the table to the client, or any other relevant criteria. Typically, a major revision is required when the entire database table should be copied to a client such as at initial creation of the table or when the overhead of applying the many updates is greater than simply copying the table.

DEPR:
Those skilled in the art will also see the ability to auto-update the client portion itself so that if a new database engine type is presented or made known to the server, at that point, the server may download a new addition of the client code that will interact with the new database engine type. Furthermore, those skilled in the art will recognize that "applets" as supported by the Java programming language may be used to implement such innovations.

DOCUMENT-IDENTIFIER: US 5997170 A
TITLE: System and method for reporting vending status
BSPR:
For example, U.S. Pat. No. 4,412,292 to Sedam et al. discloses a system that automatically communicates vending machine conditions to a central computer complex. Each vending machine includes a microprocessor which monitors, stores and transmits data to the central computer complex either immediately or at scheduled call-in times. The distributed logic between the vending machine microprocessors and central computer complex provides various alarm function signals, inventory control support and route planning for the supply and maintenance of the machines. In a preferred embodiment, the data is transmitted between the central computer complex and each vending machine over telephone lines. The system, however, requires either substantial rewiring or numerous dedicated telephone lines.

DEPR:
Vend button interface 112 monitors which vending selection button is pushed and converts the signals into a readable form. Preferably, interface 112 is adaptable to a variety of vending machine button configurations to add versatility to slave reporting unit 104 installation . That is, interface 112 is able to accurately retrieve vending status information from vending machines with varying button configurations. Upon converting retrieved vending status information, interface 112 sends the information to microcontoller 114.

DEPR:
Microcontroller 114 receives and processes vending status information signals from interface 112. Because vending status information signals may vary between vending machines, microcontroller 114 is programmed based on the type of the vending machine in which it is installed . Along with microcontroller 114 program information, vending status information received by microcontroller 114 is stored in memory 116. Configuring slave units 104 may either be accomplished by changing or updating memory 116 or by strap or switch selections. By either method, slave unit 104 is capable of collecting, processing and sending vending status information to a remote location through slave unit carrier interface 118.

DEPR:
Periodically, such as once a day, master unit 108 is programmed to either call out or wait for an incoming call from central collection office 110. Master unit 108 calls the central office at off-hours when the outside phone line is least likely needed for other purposes. Alternatively, a separate telephone line can be dedicated for these communications, in which case, unit 108 can provide more frequent updates . Central office 110 preferably uses a computer to answer and collect data from master unit 108. Central office 110 can also dial master unit 108 for vending status information. Preferably, master unit 108 is programmed to wait for a number of rings (e.g., seven) before attempting to answer. This avoids inadvertently answering other phone calls.

DOCUMENT-IDENTIFIER: US 6052053 A
TITLE: Continuous monitoring system

BSPR:
The automated systems can record all data required by documentation standards, such as ISO 9000. Reports required by such standards may be produced automatically without the need for incorporating bulky log books into the documentation or manually inputting data from log books into an electronic format. Thus, documentation processing requirements are dramatically reduced. The automated system also reduces or eliminates worker errors in recording of failures. Moreover, a supervisor may instantly confirm that each ESD device is functioning properly without having to go to each work station and review a paper log.

DEPR:
Computer 12 periodically polls data hub 14 and retrieves the event data recorded in the memory of data hub 14. Computer 12 generates a display of the status of each workstation 16. The polling rate may be very high, e.g., once per second, to generate a virtually continuous, real-time display of work station status. Of course, the polling rate may be less frequent, updating the display at any rate desired by the operator. The polling rate may be selected to reflect the monitoring requirements of the particular facility and processes. Computer 12 also stores the data for future use and manipulation.

DEPR:
The historical data may be collected and preserved for later use in one or more data archives. As shown in FIG. 12, the software may be configured to create databases which include all historical data recorded in a predetermined block of time or "shift." A standard three shift day may be used or alternative shift structures may be used depending on the user's needs. As shown in FIG. 13, the user can configure the software to automatically copy the database for each shift to a predetermined location. Because the size and number of databases can grow rapidly, it may be desirable to periodically archive the data in a new location--such as a high capacity storage device. As shown in FIG. 14, the software may be configured to periodically archive the data to a desired location.

DEPR:
Systems 10 according to the present invention, accordingly, may include a number of static control related sensors which may be connected in a number of ways to processors. The processors automatically and, if desired, continuously monitor particularized static control requirements for wrist straps and other devices. The processors then store, process, communicate and document results of the monitoring program. Although the foregoing is provided for purposes of illustrating, explaining and describing one of such systems in particular detail, modifications and adaptations to the described systems and other embodiments will be apparent to those skilled in the art and may be made without departing from the scope or spirit of the invention.

CLPR:
3. The system of claim 1 further comprising a circuit and interface for downloading data from the data hub to the computer.

CLPR:
4. The system of claim 1 further comprising a software program resident on the computer for programming the data hubs and periodically downloading and formatting test data from the data hubs.

DOCUMENT-IDENTIFIER: US 5640143 A
TITLE: Occupancy sensor and method of operating same

ABPL:

An occupancy sensor that provides improved performance by the inclusion of a microprocessor that controls the sensing transducers and processes the received signal to optimize desired detection performance. The occupancy sensor includes a quadrature detection technique and automatic sensitivity adjustment that reduce false detection caused by air flow, hallway traffic and other noise sources. A fault detection method executed periodically detects fault conditions in the sensor or sensing area caused by component failure or obstructions in front of the sensor. Transmitter power is varied depending on the size of the desired coverage area. Since various ambient parameters such as line voltage, light level, and temperature may be easily and usefully monitored, the microprocessor may use these variables to make intelligent decisions in controlling electrical loads, such as building lighting and ventilation.

DEPR:
At the start of the main loop, the sensor determines the room size and performs some elementary configuration 92. It then performs a self-test to determine that the signal path is complete 93. Next, the sensor takes two quadrature samples from the receiver 94, and runs the occupancy signal through the bandpass filter 95. The sensor then checks to see if it still has AC power 96, and then processes the output of the bandpass filter to determine if it has seen motion 97. The sensor updates its sensitivity to reflect changes in the user-controlled setting or automatic adjustments 98. Finally, the sensor will go into automatic gain setback 99 if it is time to do so, and if it is just now turning the lights on, it will perform zero-cross switching and automatic timer adjustment 100. It then goes back to the beginning of the main loop. The main loop is subject to periodic interruption in order to drive the ultrasonic transmitter. The interrupt routine is described below with reference to FIG. 27.

DEPR:
The sensor updates its sensitivity value as shown in FIG. 24 (step 98 of FIG. 15). First the sensor reads the setting of the sensitivity adjust 195. If this setting has changed since the last pass through the main loop 198 or if the program is going through the main loop for the first time 196, the sensor sets the sensitivity value to that which the user selects 197. If neither of these events has occurred and the auto-adjust timer has expired 199, then it is time to alter the sensitivity setting. If the load is still powered when the auto-adjust timer expires 200, then the load has been on for over 150% of the timeout setting without any major entry motion into the room, since the auto-adjust timer is reset when such motion occurs 181. In any space that is only supposed to be occupied briefly, such as a restroom, this probably means that the sensor is erroneously detecting room motion based on hall motion, and it is time to reduce the sensitivity greatly 202. If the room is considered to be unoccupied, and no ambient noise or hallway traffic has made the occupancy signal rise to even a fraction of the threshold level (which would reset the auto-adjust timer 187), then it is time to increase the sensitivity slightly 201. This means that the sensor will decrease its sensitivity rapidly to a point where it will not erroneously detect motion, and will then increase the sensitivity slowly and conservatively. If the sensitivity has been changed, either through a user setting or through automatic adjustment, then the auto-adjust timer is reset 203.

DOCUMENT-IDENTIFIER: US 4589081 A
TITLE: Intelligent surveillance alarm system and method

ABPL:
Stochastic means (16), responsive to the output representatives of the current operational environment periodically adjust the definition of the normal operational parameters of the subject premises. There are processor means (20) for comparing the output representative of the current operational environment with the updated definition of the normal operational parameters of the subject premises, and for detecting a deviation therebetween. Output means (22), responsive to the means for comparing and detecting, provide an output representative of the detected deviation between the output representative of the current operational environment and the normal operation parameters of the subject premises. A plurality of response devices (24), responsive to the output means for providing a specific response sequence representative of the nature and location of the detected deviation.

BSPR:

The set of detection control parameters is implemented by software logic. In a preferred embodiment, the set of detection control parameters may be modified manually by data entry, or automatically, in response to changed characteristics in the subject premises to be protected or in the normal operational environment.

DRPR:
FIG. 3 is a diagrammatic plan view of a typical anomaly detection system installation at a bank location;

DEPR:
FIG. 2 is a more detailed block diagram of one embodiment of this invention 10a, in which sensor devices 12a-12n each deliver a sensory output representative of the current operational environment of the subject premises. This sensory output is converted to a processor signal by translation means 12' and is delivered to processor means 20. Sensor devices 12a-12n may include acoustical devices, metal detectors, push buttons, entry sensors, or any combination of binary output sensor devices. In this embodiment, processor means 20 may include a microprocessor, such as a Z80 based type, or other similar processor device. Processor means 20 also receives a set of detection control parameters from detection control means 18. This series of detection and control parameters are indicative of the normal operational environment data and the particular deviations that would be indicative of an abnormal condition or disturbance. Detection control means 18 includes control means 18a and adjusting means 18b, which is responsive to the processing signal from translation means 12', and serves to periodically update the detection and control parameters in order to reflect insignificant changes in the routine operational environment while retaining sensitivity to stastically significant changes which may be indicative of a disturbance.

DEPR:
In addition to being periodically updated internally by system recognition of changed conditions, the detection control parameters may be altered manually by data entry. Manually implemented alterations may include enlarging or changing the configuration of the subject premises, or instructing the system to disregard the sensory output of a known defective or inoperative sensor component.

DEPR:
A simplified diagram of a bank installation 35 of an anomaly detection system according to this invention is shown in FIG. 3. While any type of binary sensor means may be used in conjunction with the system, the following sensor types are represented in FIG. 3:

DEPR:
The stochastic analysis is the third detection mode, and checks for a statistically significant quantity of events occurring in a particular segment of time. The events may be limited to one or more designated zones and the zones may be redefined and reconstructed by data entry. The statistical event count threshold is periodically updated to reflect routine variation in activities while maintaining sensitivity to statistically significant changes in the frequency of events within a particular zone. A surrogate sensor is assigned to the stochastic analysis output, thereby enabling it to be included as an element in one of the other detection combinations. For example, one of the "and" combinations may seek to detect a true metal detector sensor state and a statistically significant change in the activity level in the teller zone. Alternatively, a "temporal" combination may seek to detect a statistically significant flurry of activity within the safe, within several minutes of a true metal detector sensor state.

PALM INTRANET

Day : Friday
Date: 1/19/2001
Time: 08:16:53

Inventor Information for 09/241823

Inventor Name	City	State/Country
ZIESE,KEVIN J.	DRIPPING SPRINGS	TEXAS

Serial Info | Contents | Details | Atty/Agent Info | Continuity Data | Foreign Data | Invent

Search Another: Serial# [ ] Search **or Patent#** [ ] Search

(To Go BACK Use BACK Button on Your BROWSER Tool Bar)

Back to || PALM || ASSIGNMENT || OASIS || Home Page

PALM INTRANET

Day : Friday
Date: 1/19/2001
Time: 08:17:08

Inventor Name Search Result

Your Search was:

Last Name = ZIESE
First Name = KEVIN J.

Serial#	Patent#	Status	Date Filed	Title	Inventor Name
09241249 713/164	Not Issued	30	02/01/1999	METHOD AND SYSTEM FOR PROVIDING TAMPER-RESISTANT EXECUTABLE SOFTWARE	ZIESE , KEVIN J.
09241823	Not Issued	30	02/01/1999	METHOD AND SYSTEM FOR DYNAMICALLY DISTRIBUTING UPDATES IN A NETWORK	ZIESE , KEVIN J.

Inventor Search Completed: No more records to search.

Search Another: Inventor

Last Name ZIESE **First Name** KEVIN J.

Search

(To Go BACK Use BACK Button on Your BROWSER Tool Bar)

Back to || PALM || ASSIGNMENT || OASIS || Home Page